Dynamic Vehicle Routing Problems: State of the art and Prospects

Victor Pillac, Christelle Guéret, Andrés Medaglia

Dynamic Vehicle Routing Problems: State Of The Art and Prospects

Victor PILLAC *,[1,2] Christelle GUERET[1] Andres MEDAGLIA[2]

[1]*Équipe Systèmes Logistiques et de Production, IRCCyN (UMR CNRS 6597)*
École des Mines de Nantes, B.P.20722, F-44307 Nantes Cedex 3, France
[2]*Centro para la Optimización y Probabilidad Aplicada (COPA) & CEIBA*
Universidad de los Andes, Cr1 Este No.19A-10 Bogotá, Colombia

July 2010

Technical Report 10/4/AUTO
École des Mines de Nantes, France, 2010

Abstract

This scientific report summarizes the results of a literature review on dynamic vehicle routing problems. After a brief description of vehicle routing problems in general, a classification is introduced to distinguish between static and dynamic problems. Then a more precise definition of dynamism is presented, supported by example of real-world applications of such problems. Finally, a detailed study of the current state of the art in dynamic vehicle routing optimization is drawn.

Résumé

Ce rapport scientifique a été rédigé comme synthèse de travaux de recherche bibliographique sur les tournées de véhicules dynamiques. Après une brève description des problèmes de tournées en général, nous proposons une classification permettant de distinguer les problèmes statiques et dynamiques. Par la suite, une description plus détaillée de ces derniers est présentée, se basant sur des exemples tirés d'applications réelles. Enfin, une étude de l'état de l'art sur les problèmes de tournées dynamiques est développée, puis complétée par une analyse des thématiques n'ayant reçu à ce jour qu'un faible intérêt de la part de la communauté scientifique, et suceptibles de constituer des pistes intéressantes de recherche.

Resumen

Este reporte científico fue escrito como síntesis de una investigación bibliográfica sobre los problemas de ruteo de vehículo dinámicos. Después de una breve descripción general de los problemas de ruteo, se propone una clasificación permitiendo una distinción entre problemas staticos y dinámicos, los últimos siendo el objeto de una definición mas detallada, basada en ejemplos de aplicaciones reales. Por fin, se presenta un estudio del estado del arte sobre los problemas de ruteo dinámico, completada por una análisis de las temáticas que podrían constituir direcciones interesantes de investigación.

*Corresponding author: victor.pillac@emn.fr

Contents

List of Tables

List of Figures

List of Algorithms

Introduction

The *Vehicle Routing Problem (VRP)* formulation was first introduced by Dantzig and Ramser (1959), as a generalization of the *Travelling Salesman Problem (TSP)* presented by Flood (1956). The VRP is generally defined on a graph $G = (\mathcal{V}, \mathcal{E}, C)$, with $\mathcal{V} = \{v_0, ..., v_n\}$ being the set of vertices, $\mathcal{E} = \{(v_i, v_j) | (i, j) \in \mathcal{V}^2, i \neq j\}$ the arc set and $C = \left(c_{ij}\right)_{(i,j)\in\mathcal{E}}$ a cost matrix defined on $\mathcal{E}$, representing distances, travel times or travel costs. Traditionally, vertex v_0 is called *depot*, while the remaining vertices represent *clients* that need to be serviced. The VRP consists in finding a set of routes for K vehicles of identical capacity based at the depot, such that each of the vertices is visited exactly once, while minimizing the overall *routing cost*.

Beyond this classical formulation, a number of variants have been identified and studied. Among the most commonly cited are the *VRP with Time Windows (VRP-TW)*, requiring the customers to be visited during a specific time interval; the *VRP with Pick-up and Delivery (VRP-PD or PDP)*, where goods have to be either picked-up or delivered in specific amounts in each of the vertices, and its variation with time windows *(PDP-TW)*. Other variations include the Heterogeneous fleet Vehicle Routing Problem *(HVRP)* and its variants with time windows or pickup and delivery, in which the vehicles have different capacities. When transportation of people between two locations is considered, the problem is referred to as *Dial A Ride Problem (DARP)* for land transport; or *Dial A Flight Problem (DAFP)* for air transport.

In contrast with the canonical definition of the vehicle routing problem, real-world applications often include two important dimensions: *evolution* and *quality of information* (Psaraftis, 1980). Evolution of information is related to the fact that in some problems the information available to the planner may change during the execution of the routing, for instance with the arrival of new client requests or changes in the travelling times. Quality of information reflects possible uncertainty on the available data, for example when the demand of a client is only known as an estimation of its real demand. From this two dimensions, four categories of problems can be identified, as summarized in Table 1.

		Information quality	
		Deterministic knowledge of input data	Stochastic knowledge of input data
Information evolution	Input is known beforehand	Static Deterministic	Static Stochastic
	Input changes over time	Dynamic Deterministic	Dynamic Stochastic

Table 1: *Taxonomy of vehicle routing problems by information evolution and quality.*

In *static and deterministic* problems, all input is known beforehand and no change can be applied to the routing plan. This is the historical class of problem, and it includes the classical vehicle routing problem and its variations discussed earlier.

Static and stochastic problems are characterized by input partially known as random variables, of which the realization is only revealed to the planner during the execution of the routing. Additionally, only minor changes on the routing are allowed: generally a vehicle can only plan a trip to the depot, or skip a client. Applications falling in this category do not require any technological support: the planning is done before-hand

and subsequent decisions are taken autonomously by vehicles.

In *dynamic and deterministic* problems, part or all of the input is unknown and revealed dynamically during the design or execution of the routing (Psaraftis, 1995), while no stochastic data is available regarding the dynamically revealed information (Flatberg et al., 2007). In this case, the routing plan can be completely redefined in an on-going fashion. Applications in this class of problem require technological support, allowing in particular real time communication between the vehicles and the decision maker, for instance with the use of mobile phones, personal digital assistants *(PDA)* or *smart-phones*, possibly coupled with global positioning systems *(GPS)*.

Similarly, *dynamic and stochastic* problems have part or all of their input unknown and revealed dynamically during the determination or execution of the plan (Psaraftis, 1995), but in this case, exploitable stochastic knowledge is available on the dynamically revealed information (Flatberg et al., 2007). As before, the routing plan can be completely redefined in an on-going fashion, and technological support is required.

It is important to note that some variations exist among author on the classification of vehicle routing problems (Flatberg et al., 2007, Ghiani et al., 2003, Psaraftis, 1988, 1995), nevertheless the above definitions will be used as a reference in the remainder of this document.

Static and deterministic problems have been extensively studied in the literature. We refer the interested reader to the recent reviews of exact and approximated approaches by Baldacci et al. (2007), Cordeau et al. (2007b) and Toth and Vigo (2002).

As stated before, a problem is said to be *static and stochastic* if part of the input is uncertain but can be modelled as random variables of known distribution. Uncertainty may affect any of the input data. Nonetheless, as identified by Cordeau et al. (2007b), the three most studied cases are *stochastic customers*, when each customer i is present in the instance with probability p_i and absent with probability $1 - p_i$ (Bertsimas, 1988, Waters, 1989); *stochastic times*, in which case the time s_i needed to serve customer i, or the travel time t_{ij} on the edge (i, j) are random variables (Kenyon and Morton, 2003, Laporte et al., 1992, Verweij et al., 2003); and finally *stochastic demands*, with the demand ξ_i of a customer i being defined as a random variable (Christiansen and Lysgaard, 2007, Dror et al., 1989, Laporte et al., 2002, Mendoza et al., 2009, Secomandi, 2000, Secomandi and Margot, 2009).

In most cases, the realization of random variables is revealed only at operation time. For instance the actual demand of a client may only become known when it is actually visited. However, internal or external factors prohibit the enforcement of decisions in a real-time manner. Thus the objective in static and stochastic routing problems is to build a *robust* plan beforehand, that will undergo only limited changes during its execution. Therefore, in this context, the qualifier *static* does not mean that the available information or the actual routing are not subject to changes, but instead that these changes are not directly considered.

Uncertainty in the stochastic VRPs input has been addressed by various solution frameworks, of which the two most studied are the *Chance Constrained Programming (CCP)* and the *Stochastic Programming with Recourse (SPR)*. Both frameworks are based on a two-stage approach: the first phase is performed before the start of the planning horizon and aims to design a *robust* routing plan; while the second phase takes place during its execution, and consists in taking *recourse* or *corrective* actions as the realizations of the random variables are disclosed. The conceptual difference between the two approaches lies in the objective of the first-stage optimization: in CCP, the goal is to ensure an upper bound on the probability of a failure, regardless of the expected cost of the second phase; while SPR seeks the minimization of the expected cost of recourse actions. Recourse actions, on the other hand, are common to most approaches and depend on the application. For instance, for the commonly studied VRP with Stochastic Demands *(VRPSD)*, a route failure may occur any time a customer demand exceeds the associated vehicle remaining capacity. In this case, an intuitive recourse action is for the vehicle to go back to the depot to restore its initial capacity and then resume its route (Mendoza et al., 2009), or to allow the service of additional customers before returning to the depot (Novoa et al., 2006). As mentioned by Secomandi and Margot (2009), a distinction must be made between *reactive* actions, which are only taken when a failure occurs, and *proactive* actions, that are decided by anticipating the likeliness of a failure.

Further details on the static stochastic vehicle routing can be found in the reviews by Bertsimas and Simchi-Levi (1996), Cordeau et al. (2007b) and Gendreau et al. (1996).

The remainder of this document focuses on dynamic routing and is organised as follows. Section 1 introduces a general description of such problems. This section also presents the degree of dynamism and detail indicators to measure it, then reviews different applications related to dynamic routing. Section 3 provides a comprehensive survey of solution approaches, for both deterministic and stochastic dynamic routing. Finally directions for further research will be drawn, with an emphasis on the problem that we tackle in the future.

1 Dynamic routing

1.1 A general definition

To the best of our knowledge, the first reference to a dynamic vehicle routing problem is due to Wilson and Colvin (1977). They studied a dial-a-ride problem *(DARP)* with a single vehicle, in which customers requests are trip from an origin to a destination that appear dynamically. Their approach uses insertion heuristics able to perform well with low computational requirement. Later work by Psaraftis (1980) introduced the concept of *immediate request*: a customer requesting service always wants to be served as early as possible, requiring the immediate re-planning of the current vehicle route. They propose a dynamic-programming algorithm for the static case, as well as an adaptation to the dynamic context tested on a ten-customer instance.

Since then, a number of technological advances have increased the importance of *online* or *real-time* applications. With the introduction of the *Global Positioning System (GPS)* in 1996, the development and widespread use of cell-phones and smart-phones, combined with accurate *Geographic Information Systems (GIS)*, companies are now able to track and manage their fleet in real-time and cost effectively. While traditionally a two-step process, vehicle routing can now be done dynamically, introducing greater opportunities to reduce operation costs, improve customer service, and reduce environmental impact.

The most common source of dynamism in vehicle routing is the online arrival of customer requests during the operation. More specifically, a request is generally composed of a location and a demand (Attanasio et al., 2004, Goel and Gruhn, 2008, Ichoua et al., 2006, Mes et al., 2007, Mitrovic-Minic and Laporte, 2004), although in some cases, the only relevant information is the location (Beaudry et al., 2008, Bent and Hentenryck, 2005, Bertsimas and Van Ryzin, 1991, Gendreau et al., 1999, Larsen et al., 2004), or the demand (Secomandi, 2000, Secomandi and Margot, 2009, Thomas, 2007, van Hemert and Poutré, 2004). Travel time, a dynamic component of most real-world applications, has been recently taken into account (Attanasio et al., 2007, Barcelo et al., 2007, Chen et al., 2006, Fleischmann et al., 2004, Haghani and Jung, 2005, Haghani and Yang, 2007), while service time, that is, the time spent at the location of a client, has not been explicitly studied but remains considered in most approaches. Finally, some recent work considers vehicle availability (Li et al., 2009a,b), with the source of dynamism being the possible breakdown of vehicles.

1.2 Differences with static routing

By their nature, dynamic routing problems differ from their static equivalent by adding more degrees of freedom for the decision making, and introducing new metrics for the objective function.

Degrees of freedom

In some contexts, such as the pick-up of express courier studied in (Gendreau et al., 1999), the transport company does not have the obligation to service a customer request. As a consequence, it can reject a request, either because it is simply impossible to serve it, or because the cost of serving it is too high compared with the company objectives. This process of acceptance/denial has been used in many approaches (Attanasio et al., 2004, Fagerholt et al., 2009, Gendreau et al., 1999, Ichoua et al., 2000, 2003, 2006, Li et al., 2009a), and is sometimes referred to as *service guarantee* (Van Hentenryck and Bent, 2006). In some cases, the knowledge of stochastic information on customer can be used to refine the acceptance criteria. For instance in scenario-based approach (Bent and Hentenryck, 2005, Bent and Van Hentenryck, 2004a,b, 2007, Hvattum et al., 2006), artificial clients are generated to create scenarios, allowing the acceptance of a request to be conditioned on the fact that a minimum proportion of scenarios can accommodate the new request, possibly by removing up to a certain number of sampled customers. Therefore, a request is accepted or rejected not only based on the currently known requests, but also on the expected requests. A consequence is that highly restrictive requests, located for instance in remote areas, are likely to be rejected to preserve more promising expected ones.

In most of the literature, a vehicle cannot be diverted while moving toward its next destination. However, being able to redirect a moving vehicle to a new request near its current position could allow additional savings. Nevertheless, it requires real-time knowledge of the vehicle positions and to be able to communicate rapidly with drivers to assign them new destinations. Thus, this strategy has received limited interest, with the main contributions being the early work by Regan et al. (1998), the study of diversion issues by Ichoua et al. (2000), and the work by Branchini et al. (2009). However, it is worth noting that diversion has been more widely applied in the context of emergency services operations (Gendreau et al., 2001, Haghani and Yang, 2007).

Objective functions

Aside from the difference in the way input is revealed, dynamic problems also frequently differ from their static counterparts in their objective function (Psaraftis, 1995). In particular, while a common objective in static context is to minimize a routing cost, dynamic routing may introduce other notions such as service level, throughout or revenue maximization.

Having to answer dynamic customer requests introduce the notion of response time: a customer might request to be served as soon as possible, in which case the main objective may become to minimize the time between a request and its service.

Another difference is that in static planning, if not specified, all requests have the same *priority*. By contrast, in a dynamic context, nearer term events are generally more important (Psaraftis, 1995). In fact, mobilizing resources for future requests may reveal to be suboptimal if intermediate requests are to appear in the meantime. Therefore and as stated before, the decision maker might have the possibility to delay or reject customer requests. Resulting objectives could be the maximization of the number of served clients (referred to as throughout), or the total expected revenue, which may include both the routing costs and the revenue generated by the served customers.

Finally, other objectives could be introduced, for instance a pragmatic approach could be to mitigate changes in the routes when a new client request is accepted.

1.3 Measuring the dynamism

Before presenting the different applications of dynamic vehicle routing, it is important to introduce a notion that has been widely used to characterize the degree of dynamism of a problem. In fact, it appears that the level of dynamism can greatly vary between problems or even instances of a same problem. For instance, an *on-demand* transportation company may request their customers to book their trip in advance to only face limited requests for the same day. In contrast, emergency services will have to respond quickly to purely dynamic requests (Haghani and Yang, 2007).

Two dimensions can be identified to characterize the dynamism of a problem (Ichoua et al., 2006): the frequency of changes, which is the rate at which new information becomes available, and the urgencies of requests, defined as the time available between the arrival of a request and its expected service date.

Degree of dynamism

Lund et al. (1996) proposed a first metric to measure the dynamism of a problem. The *degree of dynamism* δ is defined as the ratio between the number of *dynamic* requests n_d and the total number of requests n_{tot} :

$$\delta = \frac{n_d}{n_{tot}}$$

Effective degree of dynamism

Based on the fact that the arrival time of requests is also important, as stressed by Psaraftis (1988, 1995), Larsen (2001) proposed another indicator, namely, the *effective degree of dynamism* δ^e. This metric can be interpreted as the normalized average of the arrival times. Let T be the length of the planning horizon, $\mathcal{R}$ the set of requests, and t_i the arrival time – or *disclosure date* (Jaillet and Wagner, 2006) – of request $i \in \mathcal{R}$. Considering that requests known beforehand have a disclosure date equal to 0, δ^e can be expressed as:

$$\delta^e = \frac{1}{n_{tot}} \sum_{i \in \mathcal{R}} \frac{t_i}{T}$$

Time windows

Larsen (2001) also extended the *effective degree of dynamism* to problems with time windows, in order to reflect the level of urgencies of requests. He defines the *reaction time* r_i as the difference between the arrival date t_i and the end of the corresponding time window l_i, highlighting that greater reaction times mean more flexibility to insert the request into the current routes. Thus the *effective degree of dynamism* measure is extended as follows:

$$\delta^e_{TW} = \frac{1}{n_{tot}} \sum_{i \in \mathcal{R}} \left(1 - \frac{r_i}{T}\right)$$

It is worth noting that these three metrics only take values in the interval $[0, 1]$ and increase with the *level of dynamism* of a problem. Larsen et al. (2002, 2007) use the effective degree of dynamism to define a framework classifying DVRPs among weakly, moderately and strongly dynamic problems, with values of δ^e being respectively lower than 30%, comprised between 30% and 80%, or higher than 80%.

Although the *effective degree of dynamism* and its variations have proven to capture well the time-related aspects of dynamism, it could be argued that they do not take into account the other possible sources of dynamism listed previously. In particular, the geographical distribution of requests, or similarly the traveling times between requests, are also of great importance in applications aiming at the minimization of response time. In addition, the frequency of changes mentioned earlier is not directly reflected, although it can have dramatical impact on the time available for optimization between the arrival of two events.

2 A review of applications

ITS As it has been stated before, recent advances in technology allowed the emergence of a wide new range of applications for vehicle routing. In particular, the last decade has seen the development of Intelligent Transport Systems *(ITS)*, based on the combination of geolocation technologies, with precise geographic information systems, and increasingly efficient hardware and software for data processing and operations planning. We refer the interested reader to the study by Crainic et al. (2009) for more details on *ITS* and the contribution of operations research to this relatively new domain.

AFMS Among the *ITS*, the Advanced Fleet Management Systems *(AFMS)* are specifically dedicated to the management of a corporate vehicle fleet. The problem faced is generally to deliver (pickup) goods or persons to (from) locations distributed in a specific area. Customer requests can be either known in advance or appear dynamically during the day, and the vehicles have therefore to be dispatched and routed in real-time, potentially by taking into account other factors such as traffic conditions. On top of the technological requirements, an important component of such systems is the optimization of the decision process. Traditionally, vehicle routing relied on teams of human dispatchers, meaning that the performance of the organisation as a whole was highly related to the competence and experience of the people in charge, but also that management costs were directly linked to the size of the fleet (Attanasio et al., 2007). Advances in computer science allowed the transfer of techniques from operational research to such systems, as presented in the studies by Attanasio et al. (2007), Godfrey and Powell (2002), Powell and Topaloglu (2005), Simao et al. (2009) and Slater (2002).

The remainder of this section presents more details on some of the applications where dynamic routing has been – or can be – implemented.

2.1 Visiting of clients

Applications in this category do not include the *transport* aspect of other routing problems: a request is only composed by a *customer* location and a possible time window, thus the routing plan has to ensure that each *customer* is visited, without considering capacity constraints.

Equipment maintenance A common application of dynamic routing can be found in the technical maintenance operations, for instance for copying machines, heating systems, elevators or even water meters. Companies offering such services are bound by contract to their clients, which can specify periodical, or planned maintenance visits, but may as well define a maximum delay before an intervention in the case of an equipment failure. As a consequence, at the beginning of the day, the planner or dispatcher has a set of customer requests, either from contractual visits or emergencies from the day before. Each technician is assigned a set of requests, and a routing plan is produced. However, new urgent requests appearing during the day, or changing traffic conditions in urban areas require to dynamically re-evaluate the routing of the technician fleet to ensure the best level of service. An important feature often present in such companies is the qualification of technicians for specific equipments or their assignation to geographic zones, increasing the complexity of the routing.

Traveling physician Another interesting application of dynamic routing is provided by the problem faced by a french organization: *SOS Médecins*. This non-profit organization was created to match a gap between proximity physicians who only provide limited service outside business hours, and emergency services operating 24/7 but with limited capacity for lesser urgent concerns. It operates with a fleet of physicians, in contact with a call center coordinated with other emergency services. When a person calls, the priority of the situation is evaluated, and a visit by a practitioner is planned accordingly. As for other emergency services, having an efficient dispatching system can allow a reduction in the response time, improving the level of service for the society. It requires to be able to decide the routing of the practitioner fleet in real-time, depending on the current emergencies, but also to ensure a proper coverage of the considered area, in order to have practitioners available where emergencies are likely to come up.

2.2 Transport of goods

City Logistics Due to the fact that urban areas are often characterized by travelling times varying greatly depending on the time of the day, transport of goods in such areas have led to the definition of a specific category of applications: *city logistics*. Taniguchi and Thompson (2002) defined city logistics as *the process for totally optimising the logistics and transport activities by private companies in urban areas while considering the traffic environment, the traffic congestion and energy consumption within the framework of a market economy*. Barcelo et al. (2007) developed a general framework for these applications, presenting the different modules required, from the modelling of the city road network, to the acquiring of real-time traffic data, and the dynamic routing of a fleet of vehicles in this environment.

Courier An example of application in city logistics is the courier service present in major urban areas. A case study by Attanasio et al. (2007) outlines the benefits of using computer-based techniques over human-based dispatching. They studied the case of eCourier Ltd, a London based company offering courier services. Their clients are mainly law firms, financial institutions and advertising agencies, and other entities requiring fast delivery of items or original signatures on documents. When a customer request is being known and accepted, a courier is sent to the client location to pick-up the item, that will then be delivered at its destination, in exchange of a signature. Depending on the level of service asked by the customer, the courier may consolidate or not its pick-up and deliveries, meaning that he may pick-up from various clients before delivering an item, or provide an exclusive expedited service. Companies offering courier services often have an heterogeneous fleet, composed in the case of eCourier of bicycles, motorbikes, cars and small vans. The problem is then to dynamically affect requests to couriers, depending on the type of request and its pick-up and delivery location and possible time windows, but also taking into account traffic conditions and varying travel times. This study showed that the use of an AFMS, including optimization algorithms, was highly profitable for courier companies. In fact, aside from the improvements in service quality, response time, and courier efficiency, the use of an automated system allows to increase the fleet size without the need of additional supervisors. A direct consequence is that the growth of the fleet does not require more supervisors, therefore company growth also increases the margin on each delivery, providing a strong competitive advantage.

Newspaper delivery The delivery of newspapers and magazines, as studied by Bieding et al. (2009), constitutes another highly competitive market where the satisfaction of subscribers is considered as extremely important. When a magazine or newspaper is not delivered, a subscriber can contact a call center and is offered to choose between a voucher or a subsequent delivery. In the latter case, the request is then forwarded to the delivering company, and finally transmitted to a driver that will do the proper delivery. Traditionally, this process relies on different medias, with phone calls, faxes and printed documents, and the subsequent delivery requests are not transmitted to the drivers until they come back to the depot. In response to this problem, their study proposes a centralized application, that makes use of mobile phones to communicate with drivers and *intelligently* perform the routing in real-time, to accommodate subsequent deliveries in shorter delays, reducing costs, and improving customer satisfaction.

Other applications Apart from classical routing problems on a road network, similar problematic can also be found in the internal operation of organisations. The review by Stahlbock and Voss (2008) on operations research applications in container terminals gives an interesting example of such problem, in particular with the dynamic stacker crane problem (Balev et al., 2009, Berbeglia et al., 2010), which considers the scheduling and routing of container carriers operating the loading and unloading of ships in a terminal. Other applications include transport of goods inside warehouse (Smolic-Rocak et al., 2010), factories, or even hospitals where documents or equipments have to be transferred efficiently between services (Fiegl and Pontow, 2009).

2.3 Transport of persons

The transport of persons is in general – and by many aspects – similar to the transport of goods. However, it is characterized by additional constraints, in particular regarding waiting times and travel times for the passengers that generally have to respect maximal values.

Taxi services Taxi services are the most common on-demand individual transport system. Requests from customers are composed of a pick-up location and time, possibly coupled with a destination. They can be either known in advance, for instance when a client book a cab for the next day, or arrive dynamically, in which case a taxi must be dispatched in the shortest time. When clients cannot share a same vehicle, there is only limited space for optimization, and the closest free taxi is generally the one which takes the ride. The study by Caramia et al. (2002), generalized by Fabri and Recht (2006), focuses on a multi-cab metropolitan transportation system, where a taxi can transport more than one client at the same time. In this case the online algorithms aim at minimizing the total travel distance while assigning requests to vehicles and computing the cabs routes.

On-demand transport The above mentioned multi-cab transportation systems can be generalized to other on-demand or door-to-door services. Main applications include transport of children, elderly or disabled people, or patients, for instance from their home to schools, place of work or medical centers (Lehuede et al., 2008). An extensive review of this class of problems can be found in Cordeau et al. (2007a).

Patient transport A singular application of on-demand person transportation can be found in major hospitals, with services possibly spread across various buildings on one or more campuses. Depending on the results of a consultation or surgery, or the arrival of an emergency, a patient may need to be transferred on short notice from one service to another, possibly requiring trained staff or specific equipment for its care. This application as been for instance studied by Beaudry et al. (2008).

Air taxi Air taxis services developed as a response to limitations of traditional air transport systems. Firstly, with the increase of operating costs and the constant pressure on reducing expenses, companies are increasingly reluctant to invest in private aircraft, preferring to pay for this service only when needed. Secondly, major airlines only operate from a limited number of airports, to reduce their costs first, but also because technical restrictions do not allow them to land their aircraft on smaller airports. Besides, *hub* airports often suffer from high congestion, either on the runways or inside the airport itself, thus increasing travel times. Finally, airline flight schedules are fixed and therefore do not offer a convenient alternative to the use of private jets when flexibility is needed. In contrast, air taxi companies offer an on-demand service: customers book a flight at most a few days in advance, specifying whether they are willing to share the aircraft, stop at an intermediate airport, or have flexible traveling hours. Then the company accommodates these requests, trying to consolidate flights whenever possible. Passengers can travel through smaller airports, avoiding waiting times at check-in and security checks. Between the years 2001 and 2006, the NASA conducted a study on a Small Aircraft Transportation System *(SATS)*, aiming at providing an efficient alternative to classical air transport, by making use of a new generation of cheap small aircraft (Abbott et al., 2004, Holmes et al., 2004) and relying on smaller air taxi companies. To our knowledge, the related optimisation problems have not been subject to much attention, except in the studies by Chavan (2003), Cordeau et al. (2007a), Espinoza et al. (2008a,b) and Fagerholt et al. (2009). A similar problematic can be found in the helicopter transportation systems, for instance for the transport of persons between offshore petroleum platforms (Romero et al., 2007).

Emergency services Emergency services, such as ambulances, police or fire services, constitute a domain where strong dynamism is present. Although their operation might differ, the efficiency of such services is directly linked to their response time. In other words the system has to provide guarantees between the call and arrival time. For instance, the United States Emergency Medical Services Act of 1973, requires 95% of the medical emergency calls to be treated in less than 10 minutes. The operation of emergency services is traditionally decomposed into three main aspects: the deployment of the fleet to predefined sites, to provide adequate coverage of the service area; the assignment of vehicle to emergencies, possibly by diverting an on-route vehicle; and finally, the proper routing of vehicles, possibly depending on traffic information. The survey of ambulance deployment by Brotcorne et al. (2003) reveals that this aspect has mainly been considered at the strategic level, with static approaches, but that dynamic approaches such as the one proposed by Gendreau et al. (2001) with their case study of the Montreal ambulance system, or the later work by Haghani and Yang (2007), who proposed an integrated management of emergency services, including ambulance, police and fire services, would lead to major improvements in emergency services performance.

Car pooling In a context of increasing concern on the environmental footprint of human activities, in addition to traffic issues in major urban areas, *car-pooling* has been a subject of increasing interest. Currently most systems rely on an open market place, where drivers and passengers meet and match their offer and demand for trips. An enhanced system could include a suggestion process that will find the best matches between drivers and passengers. However, the drawback most commonly cited about car-pooling is that passengers are bound to their driver and might therefore have due to unexpected delays. One could therefore imagine a dynamic car-pooling framework, where both drivers and passengers could modify their travel date at any time, with an assignment or recommendation system being in charge of finding the best matches.

3 State of the art

Few research was conducted on dynamic routing between the work of Psaraftis (1980) and the late 1990s. However, the last decade has seen an important increase in the interest for this class of problems (Eksioglu et al., 2009), with the introduction of a variety of methods ranging from linear programming to metaheuristics. The interested reader is referred to the work by Ghiani et al. (2003), Ichoua (2001), Ichoua et al. (2006), Jaillet and Wagner (2008), Larsen et al. (2008) and Psaraftis (1995) to complement our review.

3.1 Dynamic and deterministic routing

In the context of dynamic problems, critical information is revealed over time, meaning that the complete definition of an instance of a given problem is only known at the end of the planning horizon. As a consequence, an optimal solution can only be found *a-posteriori*. Therefore, most approaches use fast approximation methods that give a *good* solution in a relatively low computational time, rather than exact methods that would only provide an optimal solution for the current state, providing no guarantee that the solution will be optimal once new data becomes available.

The following paragraphs will present some approaches that have been successfully applied to dynamic routing, when no stochastic information is considered.

3.1.1 Dynamic programming

The first application of an optimization technique to dynamic routing is due to Psaraftis (1980), with the development of a dynamic programming approach. His research focuses on a dynamic program formulation to solve the static version of the dial-a-ride problem, which is then adapted to the dynamic context by running it to find the new optimal route each time a new request is made known.

It has been long known that the main drawback of dynamic programming is its computational complexity, namely the *curse of dimensionality* (see Powell, 2007, Chap. 1), therefore it has not been applied to large instances.

3.1.2 Linear programming

Rolling horizon

Yang et al. (2004) addressed the real-time Multivehicle Truckload Pickup and Delivery Problem by introducing two reoptimization strategies: *MYOPT* and *OPTUN*. Their studied problem is based on a company with a fleet of K trucks that has to serve point to point transport requests arriving dynamically. Important assumptions are that each truck can only carry one request at a time, with no possible pre-emption, and moves at the same constant speed. Both reoptimization strategies are based on a Linear Program *(LP)* that is solved whenever a new request arrives. However, while *MYOPT* only uses deterministic costs, *OPTUN* introduces an additional *opportunity cost* on each of the arcs that is directly linked to the spatial distribution of customer requests. Consequently, the optimization will tend to reject isolated jobs, and avoid arcs that are far from potential requests.

Dynamic column generation Exploring a new area of linear programming, Chen and Xu (2006) designed a dynamic column generation algorithm (*DYCOL*) for the dynamic version of the classical Vehicle Routing Problem with Time Windows *(D-VRPTW)*. The authors define K *decision epochs* over the planning horizon, corresponding to the dates when the solution is re-evaluated along with the newly available requests. The innovation consists in dynamically generating columns for the linear model, by iterating over a four step algorithm at each decision epoch. The initialization of the algorithm is done by generating a set of routes Π_0 covering all the currently accepted requests and based on the columns used at the previous decision epoch. During the first step, a set-partitioning model is build up and its linear relaxation solved to retrieve the optimal dual values that will be used to calculate the reduced costs in the next steps. In the second and third steps, a predetermined number of new columns are generated by applying a two-phase local search to the elements of Π_0. The last step consists in generating columns for the newly arrived requests, before updating the formulation of the first step and starting a new iteration. The authors compared this new approach to a traditional column generation with no time limit (*COL*), solved to termination with the data available at each epoch. Computational results based on the Solomon benchmark (Solomon, 1987) demonstrate that the *DYCOL* procedure yields comparable results in terms of objective function, but with running times limited to 10 seconds, against up to various hours for *COL*.

3.1.3 Meta-heuristics

As it was stated earlier, the optimal solution at a time t is biased by the knowledge of the input at this same period. Therefore, enforcing a decision that is optimal at time t may reveal to be sub-optimal at time $t + 1$. Hence *exact* methods are not necessarily best fitted in the context of dynamic routing, given that approximated solution could reveal to be better when new data becomes available.

As a consequence, most of the research in the field of dynamic vehicle routing problem is based on heuristics in the general sense, including local search procedures, tabu search and evolutionary algorithms, as they will be presented in the following paragraphs.

Tabu search To the best of our knowledge, the first application of a tabu search procedure to a dynamic routing problem is due to the study by Gendreau et al. (1999). It was motivated by the local operation of long distance express courier services, which can be seen as a vehicle routing problem with time windows (*VRP-TW*). Their approach consisted in the adaptation of the framework introduced by Taillard et al. (1997) to a dynamic context. The general idea is to maintain a pool of *good* routing plans – the *adaptive memory* – which is used to generate initial solutions for a parallel tabu search. The parallelization of the search is done by partitioning the routes of the current solution, each subset being optimized by an independent thread. Whenever a new customer request arrives, it is checked against all the solutions from the adaptive memory with rapid insertion heuristics to decide whether it should be accepted or rejected. Once accepted, the tabu search is resumed including the new request. It is worth noting that because the current routing is subject to change at any time, vehicles do not know their next destination until they finish the service of a request. This framework was also implemented for the *D-VRP* (Ichoua et al., 2003) and for the *PDP* (Ichoua et al., 2000), while other variations of tabu searches have been applied to the *D-PDP* (Barcelo et al., 2007, Chang et al., 2003), or to the the *DARP* (Attanasio et al., 2004, Beaudry et al., 2008).

VNS The main idea behind Variable Neighborhood Search *(VNS)* is to iteratively improve a solution by considering different neighborhoods successively. An overview of the approach is given by Algorithm 1. The algorithm starts with an initial solution $\mathbb{X}$ and generates a neighbor $\mathbb{X}^t$ from the current neighborhood ($\mathtt{shake}$ line 6), which is then improved by a local search procedure ($\mathtt{ls}$ line 7). If the new solution is accepted ($\mathtt{accept}$ line 8), it replaces the current solution and a new iteration is performed with the first neighborhood (line 10); otherwise, the next neighborhood is selected (line 12) and a new iteration is done with the unchanged current solution. As for other heuristics, iterations are performed until a *stop criterion* is met, usually a maximum time or number of iterations. VNS was originally developed to tackle static combinatorial problems (Hansen and Mladenovic, 2001, Mladenovic and Hansen, 1997), yet its low computational times has allowed applications in dynamic contexts, with for instance the work of Goel and Gruhn (2008) on a *Generalized Vehicle Routing Problem (GVRP)*.

Algorithm 1 The Variable Neighbourhood Search generic algorithm

Input: $\mathbb{X}$ a valid solution, z an evaluation function and $\mathcal{N} = \{\mathcal{N}_1, .., \mathcal{N}_K\}$ a set of neighborhood structures
Output: $\mathbb{X}^*$ the best solution found
1: **function** VNS($\mathbb{X}$)
2: $\quad \mathbb{X}^* \leftarrow \mathbb{X}$;
3: $\quad$ **repeat**
4: $\quad\quad k \leftarrow 1$; ▷ Select first neighborhood
5: $\quad\quad$ **repeat**
6: $\quad\quad\quad \mathbb{X}^t \leftarrow \mathtt{shake}(\mathcal{N}_k, \mathbb{X})$; ▷ Generate a neighbour from neighbourhood $\mathcal{N}_k$
7: $\quad\quad\quad \mathbb{X}^t \leftarrow \mathtt{ls}(\mathbb{X}^t)$; ▷ Local search to improve $\mathbb{X}^t$
8: $\quad\quad\quad$ **if** $\mathtt{accept}(\mathbb{X}^t, \mathbb{X})$ **then** ▷ $\mathbb{X}^t$ is accepted as current solution
9: $\quad\quad\quad\quad \mathbb{X} \leftarrow \mathbb{X}^t$; ▷ Update current solution
10: $\quad\quad\quad\quad k \leftarrow 1$; ▷ Restart from first neighbourhood
11: $\quad\quad\quad$ **else**
12: $\quad\quad\quad\quad k \leftarrow k + 1$; ▷ Go to next neighbourhood
13: $\quad\quad\quad$ **end if**
14: $\quad\quad\quad$ **if** $z(\mathbb{X}^t) < z(\mathbb{X}^*)$ **then** ▷ An improvement has been found
15: $\quad\quad\quad\quad \mathbb{X}^* \leftarrow \mathbb{X}^t$; ▷ Update best solution
16: $\quad\quad\quad$ **end if**
17: $\quad\quad$ **until** $k = K$;
18: $\quad$ **until** $\mathtt{stopCriterion}()$;
19: $\quad$ **return** $\mathbb{X}^*$;
20: **end function**

LNS Large Neighborhood Search *(LNS)* was first introduced by Shaw (1998) and can be seen as a special case of VNS (Bent and Hentenryck, 2006). Algorithm 2 presents the general structure of LNS. The exploration is done by firstly *destroying* the current solution ($\mathtt{destroy}$ line 6) and then *repairing* it ($\mathtt{repair}$ line 6). The resulting neighbour $\mathbb{X}^n$ is then accepted or rejected depending on an implementation specific criterion ($\mathtt{accept}$ line 7). In the original implementation by Shaw (1998), the $\mathtt{destroy}$ method removes a subset of clients from the routes, while the $\mathtt{repair}$ function reinserts them by using heuristics and constraint propagation. On the other hand, it uses a simple acceptance criterion, accepting only improving solutions. The main idea behind LNS is to efficiently explore an exponential neighborhood. In fact, with n clients, there are 2^n candidate subsets to be removed and reinserted. In the implementation by Shaw (1998), this is achieved by the $\mathtt{destroy}$ function that considers subsets of iteratively incremented cardinality, increasing the neighborhood size each time a given number of consecutive non-improving neighbors are found[1]. After being successfully applied to static routing problems (Bent and Hentenryck, 2006, Bent and Van Hentenryck, 2004c, Pisinger and Ropke, 2007, Ropke and Pisinger, 2006, Rousseau et al., 2002), large neighborhood search was adapted to dynamic routing, either directly as in the study by Goel and Gruhn (2005) on the *D-PDP*, or as an optimization subroutine of other algorithms such as the *MSA* (Van Hentenryck and Bent, 2006).

Algorithm 2 The Large Neighbourhood Search generic algorithm

Input: $\mathbb{X}$ a valid solution, z an evaluation function
Output: $\mathbb{X}^*$ the best solution found

function LNS($\mathbb{X}$)
  $\mathbb{X}^* \leftarrow \mathbb{X}$;
  $i \leftarrow 0$ ▷ Number of iterations
  $k \leftarrow 0$ ▷ Number of non-improving iterations
  repeat
    $\mathbb{X}^t \leftarrow \mathtt{repair}(\mathtt{destroy}(\mathbb{X}, i, k))$; ▷ Generate a neighbor
    if $\mathtt{accept}(\mathbb{X}^t, \mathbb{X})$ **then** ▷ $\mathbb{X}^t$ is accepted as current solution
      $\mathbb{X} \leftarrow \mathbb{X}^t$; ▷ Update current solution
    end if
    if $z(\mathbb{X}^t) < z(\mathbb{X}^*)$ **then** ▷ An improvement has been found
      $\mathbb{X}^* \leftarrow \mathbb{X}^t$; ▷ Update best solution
      $k \leftarrow 0$; ▷ Reset the number of non-improving iterations
    else
      $k \leftarrow k + 1$; ▷ Update the number of non-improving iterations
    end if
    $i \leftarrow i + 1$;
  until $\mathtt{stopCriterion}(i, k)$;
  return $\mathbb{X}^*$;
end function

MPA Multiple Plan Approach *(MPA)* was introduced by Bent and Van Hentenryck (2004b) to tackle the *D-VRPTW*, and is presented as a generalization of the tabu search with adaptive memory developed by Gendreau et al. (1999). As illustrated in Figure 1, the general idea is to populate and maintain a pool of routing plans that are used to generate a distinguished plan[2]. Time between events is used to continuously improve the plans of the pool. Whenever a new request arrives, a procedure is called to check whether it can be serviced or not, for instance by checking if at least one plan can accommodate it. If the answer is positive, the request is inserted in the plans of the pool, while incompatible plans are discarded. Pool updates are performed periodically, or whenever a vehicle finishes the service of a customer. This phase is crucial and ensures that all plans are coherent with the current state of both vehicles and customers requests. The pool can be seen as an *adaptive memory* (Taillard et al., 2001) and allows

[1]Therefore LNS can be interpreted as a special case of VNS, considering that each cardinality defines a neighborhood.
[2]The distinguished plan is not necessarily the *best* plan of the pool (Bent and Van Hentenryck, 2004b)

the maintaining of a variety of solutions, the goal being to have a variety of plans to choose from when a new requests arrives.

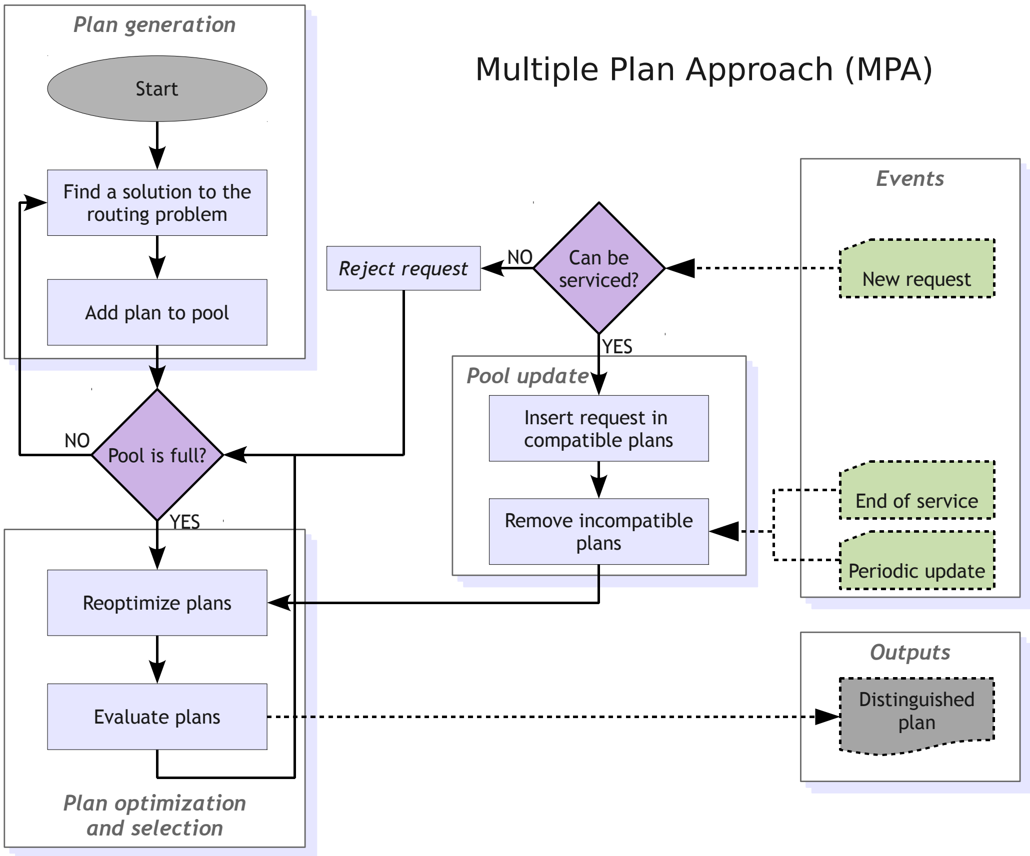


Figure 1: *An overview of the Multiple Plan Approach* (MPA) *framework. Plan generation, reoptimization and selection can be interrupted by incoming events, and are then resumed.*

Evolutionary algorithms Evolutionary algorithms have not received much interest in the field of dynamic routing until recently. In an early work, Benyahia and Potvin (1998) studied the *D-PDP* and proposed a Genetic Algorithm *(GA)* modeling the decision process of a human dispatcher. More recently, GA were also used for the same problem (Cheung et al., 2008, Haghani and Jung, 2005), and for the *D-VRP* (van Hemert and Poutré, 2004). Genetic Algorithms in dynamic contexts are very similar to those designed for static problems: a population of individuals representing routing plans is maintained, and subject to crossovers, mutations and selection. The main difference is that the GA is generally running throughout the planning horizon, and the solutions are constantly adapting to the changes made to the input.

Ant colony Montemanni et al. (2005) developed an Ant Colony System *(ACS)* to solve the dynamic vehicle routing problem. Their approach makes use of the *time slices*, as introduced by Kilby et al. (1998), that divide the day in n_{ts} periods of equal duration $\frac{T}{n_{ts}}$. The processing of a request arriving during a time slice is postponed until the end of it, thus the problem solved during a time slice only considers the requests known at its beginning. Hence the optimization is run independently during each of the time slice, resolving a static problem. The main advantage of this partition of time is that similar computational effort are allowed for each time slice, while if optimization was performed on the arrival of each new request there will be no guarantee on the time available for computation. It is made possible by the nature of requests that are never *urgent*, and which processing can therefore be postponed, in contrast with other studies (Gendreau et al., 1999).

3.2 Dynamic and stochastic routing

As presented before, stochastic and dynamic routing problems can be seen as an extension of their deterministic counterparts, where additional knowledge is available on the dynamically revealed input, under the form of stochastic information.

Sampling v.s. pricing

From this observation, two main strategies can be identified in the stochastic and dynamic routing literature, that will be referred to as either based on *sampling* or *pricing*.

As their name suggests, *sampling* strategies incorporate the stochastic knowledge by producing various *scenarios* or *simulations* resulting from the sampling of the input variable distributions, the goal being to capture the likeliness of an event and create a routing plan that will be able to accommodate it. Generally speaking, such a method will generate sampled requests and find a route servicing them with a method from *deterministic* routing. This is for instance the case of the Mutliple Scenario Approach (MSA) (Van Hentenryck and Bent, 2006).

On the other hand, approaches based on *pricing* integrate stochastic knowledge by evaluating expected values, for instance the expected traveled distance when servicing a customer with a given vehicle, knowing the probability of seeing the arrival of a new request in the meantime requiring a deviation from the initial plan. Markov Decision Process (MDP) based approaches (Novoa and Storer, 2009, Secomandi and Margot, 2009) or Approximate Dynamic Programming (ADP) (Powell, 1988, 1996) are examples of such strategies.

Generally speaking, the advantage of sampling is its simplicity of implementation, while its drawback is that a large number of samples is required to effectively reflect the reality. Alternatively, pricing strategies are often more complex in their formulation and require to compute efficiently possibly complex expected values, but they capture more formally the stochastic nature of the problem.

Examples of these two strategies, as well as other variations, will be given in the following paragraphs.

3.2.1 Markov processes

Early research by Powell et al. (1988) on dynamic routing formulated the problem as a Markov process. Unfortunately, the state and action space tends to grow exponentially as the dimension of the problem increases, inducing prohibitive computational times. Besides such modeling requires simplification assumptions that are incompatible with real world applications. Nonetheless, it allowed new insights in the field of dynamic programming, as it will be presented in the next paragraph.

Markov decision process was for instance used by Novoa and Storer (2009), Secomandi and Margot (2009) to model and solve a *Vehicle Routing Problem with Stochastic Demands (VRP-SD)*, where the set of customers is known but with random demands. Although the authors do not explicitly state it, the use of a *reoptimization* policy classifies the problem as *dynamic* in our taxonomy.

3.2.2 Approximate dynamic programming

The work by Powell (1988, 1996) opened a new branch in the area of dynamic vehicle routing by introducing the notion of predictive algorithms, which take advantage of the stochastic knowledge of the future to take predictive or preventive actions. In his early work, Powell (1988) addressed the issue of anticipatory relocation of vehicles based on prediction of future demands. His approach considered a time-space graph, with nodes for known and forecasted requests. At each period, the problem is solved by minimizing the deterministic cost of the current period, and the expected cost of future periods. Powell (1996) later furthered this research by considering the driver-request assignment aspect. The problem is modelled as a dynamic-stochastic network composed of an assignment and a forecast component, with the cost of possible recourse actions being included in the arc costs. Consequently the initial problem is approximated as a transportation problem in the resulting network, which is then efficiently solved by a network algorithm.

ADP Since then, different research has been led on this field, in particular by Powell and the CASTLE laboratory who developed a new optimization framework now known as Approximate Dynamic Programming *(ADP)*. In dynamic programming, one defines a value function at each decision step, and then generally uses a backward induction to determine the optimal actions at each decision time. However, this method suffers from the *curse of dimensionality*, making it useless when the state or action spaces are too vast. The general idea of ADP is to work in a forward fashion, avoiding the evaluation of all possible states by approximating the value function. We refer the interested reader to Powell (2009, 2007) for a more detailed description of the framework.

During the last decade, ADP has been successfully applied to fleet management problems (Godfrey and Powell, 2002, Powell and Topaloglu, 2005, Simao et al., 2009), freight transport (Powell et al., 2007, Powell and Topaloglu, 2003), with in particular an application to the VRPSD (Novoa and Storer, 2009), but also dynamic assignment problem, which extend the fleet management (Powell, 1996, Spivey and Powell, 2004), or other dynamic flow problems (Topaloglu and Powell, 2006). As in Powell (1996), ADP proceeds by defining the problem at each decision time using a state-action graph representation, adding the information given by the approximated value function as costs on specific arcs, and using a graph algorithm to solve the resulting flow problem.

3.2.3 Online stochastic combinatorial optimization

The other main branch in predictive approaches for the online VRP is the work done by Bent and Van Hentenryck (Bent and Hentenryck, 2005, Bent and Van Hentenryck, 2004a,b,c,d, Hentenryck et al., 2010, Van Hentenryck and Bent, 2006) who developed generic frameworks and algorithms for the *D-VRP* under the name Online Stochastic Combinatorial Optimization *(OSCO)* with contributions and adaptations from the area of online scheduling. Figure 2 presents an overview of the research done in this area.

MSA Multiple Scenario Approach *(MSA)*, of which an overview is given in Figure 3, was first presented as a predictive adaptation of the MPA framework discussed earlier (Bent and Van Hentenryck, 2004b). The main idea is to include stochastic knowledge of the problem by considering possible future requests and by maintaining a pool of *scenarios*, each one composed by a tour visiting all actual nodes and a set of virtual requests. Scenarios are build in three steps, as illustrated in Figure 4. Firstly virtual requests are generated by sampling the clients distributions; then an optimization procedure finds a *good* tour that serves both real and sampled requests; finally, sampled requests are removed from the tour. This process introduces room in the resulting scenarios that could then be used to accommodate future requests, improving the robustness of the routing, as illustrated in Figure 4. While the optimal tour for the currently known requests would start by areas where customers are likely to appear in the afternoon, the generated scenario guides the routing toward regions with early clients first, leaving for

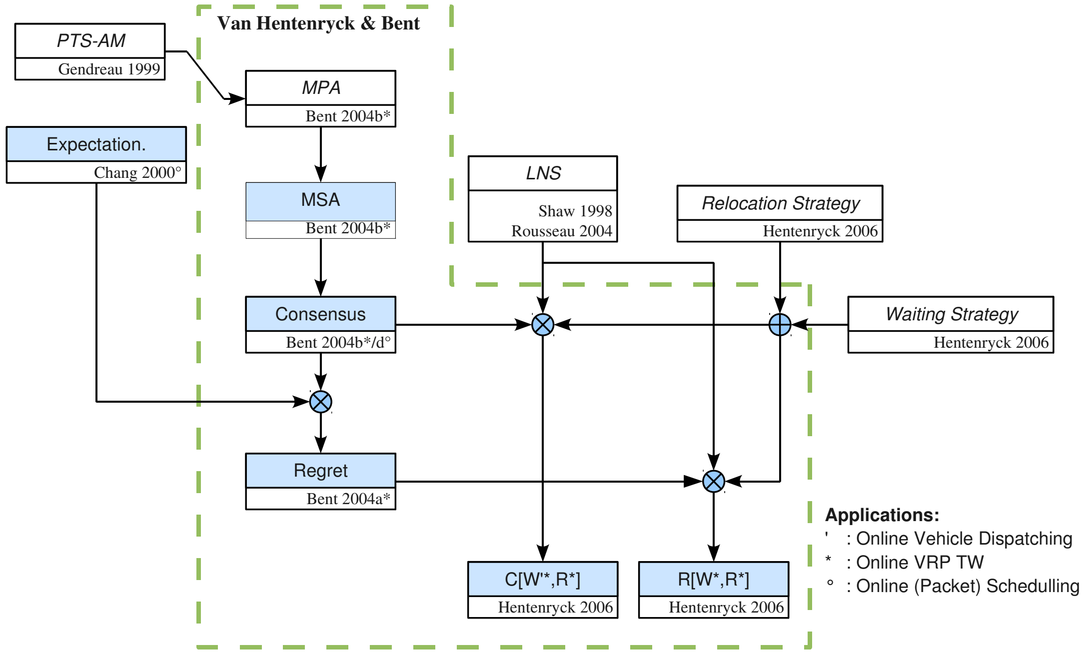


Figure 2: *Online Stochastic Combinatorial Optimization* (OSCO) *research overview for the VRP.*

instance room to accommodate potential mid-day requests between clients C and D.

Immediate decision making Another idea behind the MSA scheme is to take advantage of the time between decisions to continuously improve the current pool of scenarios. During the initialization of the algorithm, a first set of scenarios is created based on the requests known beforehand, that will then be re-optimized and completed with new scenarios during the *idle* time of the dispatcher. When a decision is required (*Decision* event in Figure 3), the *scenario optimization* procedure is suspended, and the scenario pool is used to find the best routing plan. Scenarios present in the pool that are incompatible with the resulting routing are discarded, and the *continuous improvement* algorithm is resumed.

Computational experimentations on instances adapted from the Solomon benchmark (Solomon, 1987) showed that MSA outperforms MPA both in terms of served customers and traveled distances, especially for instances with high degrees of dynamism.

We will now briefly present three predictive algorithms, which were originally designed for an online job scheduling where the goal is to select a job – or request – r^* to be scheduled at the current time t, from the set of feasible requests $\mathcal{R}$ (Hentenryck et al., 2010, Van Hentenryck and Bent, 2006). Let $\mathcal{O}$ be an optimization algorithm for the considered problem, for instance an implementation of VNS (Van Hentenryck and Bent, 2006), $z(\gamma)$ a function evaluating the *profit* of a solution – or plan – γ defined as a permutation of the n pending requests $(\gamma_0, ..., \gamma_{n-1})$, and $f^{\cdot}(r)$ an evaluation function for the benefit of scheduling request r first. The selected request r^* is formally defined as:

$$r^* = \arg\max_{r\in\mathcal{R}} f^{\cdot}(r)$$

Expectation Chang et al. (2000) designed a first selection process for online scheduling based on sampling, referred to as *expectation algorithm* (Bent and Van Hentenryck, 2004a,d, Hentenryck et al., 2010, Van Hentenryck and Bent, 2006). The evaluation of the expected profit for r is done by solving the scheduling problem defined by the combination of

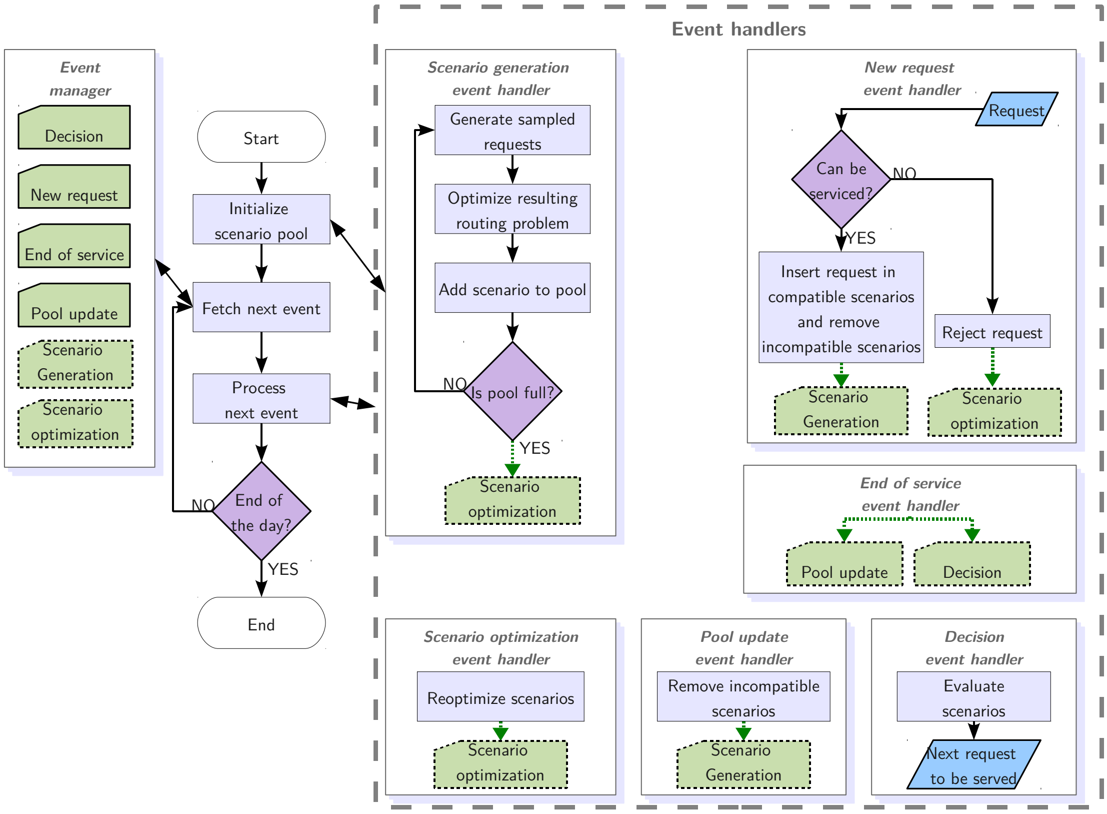
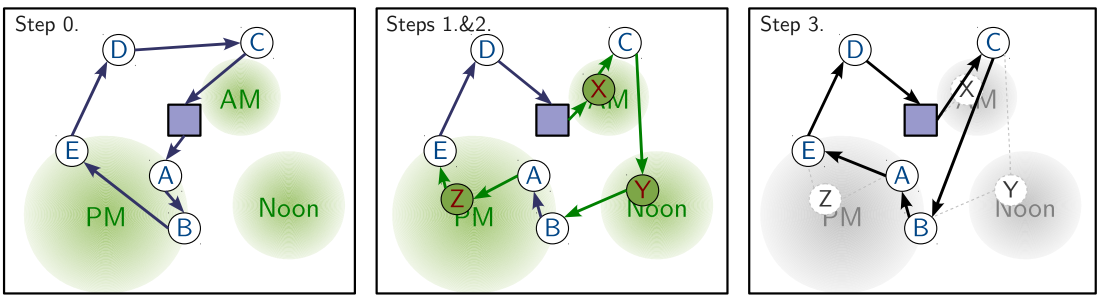


Figure 3: *An overview of the Multiple Scenario Approach* (MSA) *framework from an event-driven perspective. New events are ordered by priority and treated by the corresponding event handler. Events with continuous border are preemptive and cause the suspension of the handling of any lower priority event.*



Actual request and *a-priori* optimal tour. | Actual and sampled requests, and tour resulting from the optimization. | Resulting Scenario: actual requests, final tour and sampled requests.

Figure 4: *Scenario generation in the Multiple Scenario Approach. The AM, Noon and PM areas represents regions where clients are likely to appear during the corresponding periods of the day. The* a-priori *optimal tour of step 0. is not actually calculated and would be the result of a myopic optimization.*

pending and sampled requests from the set of *scenarios* $\mathcal{S}$, imposing that r is scheduled first:

$$f^{\mathcal{E}}(r) = \sum_{s \in \mathcal{S}} z\left(\mathcal{O}(r, s)\right)$$

The main drawback of this method is that the optimization needs to be performed $|\mathcal{R} \times \mathcal{S}|$ times, for each pair request-scenario, which can be restrictive when the set of feasible requests grows and the computational time is limited, as it is the case for online vehicle routing (Van Hentenryck and Bent, 2006).

Consensus In the *consensus algorithm* (Bent and Van Hentenryck, 2004b,d), optimization is only performed once for each of the generated scenarios. Consensus is obtained as follows: a request r is granted a profit of 1 each time r is scheduled first on a plan γ_s resulting from the optimization $\mathcal{O}(s)$ of scenario s :

$$f^{C}(r) = \sum_{s \in \mathcal{S}} y_{r,s}$$

Where $y_{r,s} \equiv (r = \gamma_{s0})$, with γ_{s0} being the first request of plan γ_s. It is important to notice that only $|\mathcal{S}|$ calls to the optimization procedure are required to calculate all the γ_s. Therefore, the main advantage in comparison with the expectation algorithm is that more scenarios can be evaluated in the same computational time. On the other hand, one of its flaws is the elitism of the selection process, which exclusively considers the first scheduled request, while ignoring requests that might be scheduled later in all scenarios, but still be robuster overall.

In (Bent and Van Hentenryck, 2004a), the authors propose an approach combining the advantages of both expectation and consensus algorithms: the consensus is used to identify a subset of promising requests, which are then compared against all scenarios in the expectation algorithm.

Regret In further work, Bent and Van Hentenryck (2004a) proposed another predictive approach combining the strengths of expectation and consensus: the *regret algorithm* (Bent and Van Hentenryck, 2004a, Hentenryck et al., 2010, Van Hentenryck and Bent, 2006). The general idea is to firstly perform an optimization for each scenario, yielding the γ_s, to then estimate for each scenario the degradation of the profit function incurred if a suboptimal request r is scheduled in place of γ_{s0}. This lost of profit is called *regret*, and it is used to approximate the benefit $z(\mathcal{O}(r, s)$ of the expectation algorithm, which exact calculation has been shown to have a prohibitive computational time.

Let $\mathtt{Regret}(r, s, \gamma_s)$ by the previously mentioned *regret* value associated with the scheduling of request r in place of γ_{s0} for scenario s. It is important to note that, in our notation, $\mathtt{Regret}(\gamma_{s0}, S, \gamma_s) = 0$. The overall profit of request r is defined as:

$$f^{R}(r) = \sum_{s \in \mathcal{S}} [z(\gamma_s) - \mathtt{Regret}(r, s, \gamma_s)]$$

As for the consensus algorithm, the optimization is only performed on $|\mathcal{S}|$ scenarios, which is a guarantee of performance if optimization is the most time consuming task. Nevertheless, an important requirement is that the evaluation of the $\mathtt{Regret}(r, s, \gamma_s)$ function has to be efficient, otherwise the flaw of the expectation algorithm will reappear. A possible definition of the $\mathtt{Regret}(r, s, \gamma_s)$ function would be to return the value 0 is there is a feasible swap between r and γ_{s0}, and a value 1 otherwise (Van Hentenryck and Bent, 2006).

Multiple vehicles These three algorithms have been described for the case of online scheduling. The generalization to multiple vehicle routing raises a number of issues (Van Hentenryck and Bent, 2006). In particular the routing has to be performed whenever at least one vehicle is available, and decisions for a single vehicle influence and are influenced by decisions for the rest of the fleet. Another characteristic is that decisions are now tuples of K requests $(r_1, ..., r_K)$ corresponding to the next request to be assign to each of the K vehicles. As a consequence, a routing plan γ_s for scenario s is now defined as a tuple of routes $(\gamma_{s1}, ..., \gamma_{sK})$, where each γ_{sk} is a route $(\gamma_{sk0}, ..., \gamma_{skn})$ associated with vehicle k.

Multiple pointwise decisions Van Hentenryck and Bent (2006) extended the consensus and regret algorithms to multiple vehicle problems by introducing a two step procedure named *multiple pointwise decisions*. The first step consists in the evaluation of each decision across all plans, similarly to the previous versions, and independently of the associated vehicle. In the second step, each plan is evaluated according to its own decisions, and the best plan is selected. More formally, the first produces values for $f(r)$ for all requests $r \in \mathcal{R}$, and each plan γ_s is evaluated by the function g defined as follows:

$$g(\gamma_s) = \sum_{k=1}^{K} f(\gamma_{sk0})$$

Where γ_{sk0} is the first request served by vehicle k in plan γ_s. Finally, the best plan $\gamma^* = \arg\max_{s \in \mathcal{S}} g(\gamma_s)$ is selected and each vehicle k that is currently available is affected to request γ^*_{k0}. It is important to note that the distinguished plan is in this case directly selected from the pool, for a plan composed by the *best* requests for each vehicle might not respect problem specific constraints, while the plan γ^* does by construction.

We refer the interested reader to the book by Hentenryck et al. (2010) for an *in-depth* presentation of the *OSCO* framework.

3.2.4 Other work

Aside from *ADP* and *OSCO*, other research has been carried on stochastic dynamic problems, mainly by adapting *deterministic* methods by introducing *sampling* or *pricing*.

DSHH Another approach similar to the *consensus* algorithm for VRP was proposed by Hvattum et al. (2006) under the name of *Dynamic Sample Scenario Hedge Heuristic (*DSHH*)*. This method is based on the division of the planning horizon into time intervals, at the beginning of which the routing is revised. Routing optimization is done as follows: firstly, a set of scenarios is generated by sampling the customer distributions. In a second step, a subset $\mathcal{P}$ of promising requests is identified by iteratively solving the scenarios, imposing that all requests from $\mathcal{P}$, initially empty, have to be served during upcoming time interval. When all scenarios have been solved, the request that is the most frequently visited during upcoming time interval is added to $\mathcal{P}$, and a new iteration is performed, unless a stop criterion has been matched. The final step performs the actual planning of requests from $\mathcal{P}$, thanks to another iterative process, by successively adding them to vehicle routes, depending on the frequency of this assignment over all the scenarios.

Local search Various local search approaches have been developed for the stochastic and dynamic problems. Ghiani et al. (2009), developed a sampling-based algorithm for the *D-PDP* where sampling is done only for the near future in order to reduce the computational effort. The main difference with MSA is that no scenario pool is used, and the selection of the distinguished plan is based on the expected penalty of accommodating requests in the near future.

Tabu search Tabu search as also been adapted to dynamic and stochastic problems, with for instance the research of Ichoua et al. (2006) for the *D-VRPTW* or Attanasio et al. (2007) for an application of the *D-PDP*, both adaptations being based on the *sampling* of the requests distributions.

Linear programming As presented in paragraph 3.1, Yang et al. (2004) proposed a LP formulation for the *PDP* that adds *opportunity costs* to the valuation of arcs, in order to reflect the expected cost of going on isolated areas. In later work, Yang et al. (2005) studied the *emergency vehicle dispatching and routing* and proposed a mathematical formulation later used by Haghani and Yang (2007) on a similar problem.

3.2.5 Additional strategies

In addition to the general frameworks described in the previous paragraphs, the introduction of probabilistic knowledge of the future allows the design and implementation of new strategies, that aim at producing a more adequate response to possible upcoming events.

Waiting strategy Waiting strategy consists in deciding the period of time a vehicle will wait after servicing a request before heading toward its next customer, or planing a waiting period on a strategic location. This strategy is particularly important in problem with time windows, where lag can appear between the servicing of two requests. Mitrovic-Minic et al. (2004) proved that in all cases it is better to wait after servicing a customer than to drive directly, but that more refined strategy can lead to further improvements in terms of travelled distances and serviced customers. The problem is in general to evaluate the probability of a new request in the neighbourhood of a serviced request, and to plan a waiting period accordingly. It has been implemented in various frameworks for the *D-VRP* (Branke et al., 2005, Thomas, 2007) and *D-VRPTW* (Bent and Van Hentenryck, 2007, Branchini et al., 2009, Ichoua et al., 2006, Van Hentenryck and Bent, 2006), the *D-PDP* (Ghiani et al., 2009, Mitrovic-Minic et al., 2004) or the *Resource Allocation Problem (RAP)* (Godfrey and Powell, 2002), and have shown to bring significant improvements, especially in the case of a limited fleet facing high new request rate.

Relocating strategies In addition to the waiting after or before servicing a client, a vehicle can be moved to a strategic location, in the neighborhood of which new requests are likely to arrive. This strategy is the keystone of emergency fleet deployment, also known as *Emergency Vehicle Dispatching – or Redeployment – Problem*, with for instances the studies by Gendreau et al. (2001) and Haghani and Yang (2007). It has also been applied to other vehicle routing problems, such as the *D-VRP* (Larsen, 2001) and *D-VRPTW* (Bent and Van Hentenryck, 2007, Branchini et al., 2009, Ichoua et al., 2006, Van Hentenryck and Bent, 2006), the *D-TSPTW* (Larsen et al., 2004), the *D-PDP* (Ghiani et al., 2009), or the Resource Allocation Problem (*RAP*) (Godfrey and Powell, 2002).

Request buffering Request buffering was introduced by Pureza and Laporte (2008). It consists in differing the assignment to vehicles of some requests, by storing them in a buffer, allowing more *urgent* requests to be treated first. It introduces more flexibility in the *service guarantee* by postponing the assignment decision.

3.3 Performance evaluation

In contrast with static problems where measuring the performance of an algorithm is straightforward and consists in comparing running times or results, dynamic contexts require the introduction of new metrics to prove analytical properties or compare empirical results, and assess the performance of a particular method or algorithm.

Competitive ratio

Sleator and Tarjan (1985) were the first to introduce a now commonly used metric, at least as far as theoretical performance is concerned: the *competitive analysis* (Jaillet and Wagner, 2008, Larsen et al., 2007). Consider a minimization problem P and $\mathcal{I}$ the corresponding set of possible instances. Let $z^*(I_{of})$ be the optimal cost for the *offline instance* I_{of} corresponding to $I \in \mathcal{I}$. By *offline instance* we mean that in I_{of} all the input data from instance I, either static or dynamic, is available when building the solution. In contrast, the data of the *online* version is revealed in a real-time fashion, thus an algorithm $\mathcal{A}$ has to take into account new information *on-the-go*, and produce a solution relevant to the current state of knowledge. Let $z_{\mathcal{A}}(I) = z(x_{\mathcal{A}}(I))$ be the cost of the final solution $x_{\mathcal{A}}(I)$ found by the on-line algorithm $\mathcal{A}$. Algorithm $\mathcal{A}$ is said to be *c-competitive*, or equivalently to have a *competitive ratio* of c, if there is a constant α such that

$$z_{\mathcal{A}}(I) \leqslant c \cdot z^*(I_{of}) + \alpha \quad , \quad \forall\, I \in \mathcal{I}$$

In the case where $\alpha = 0$, the algorithm is said to be *strictly-c-competitive*, meaning that in all cases the objective value of the solution found by $\mathcal{A}$ will be at most of c times the optimal value. The competitive ratio metric allows a worst-case absolute measure of an algorithm performance in terms of objective value. We refer the reader to Borodin and El-Yaniv (2005) for an in-depth analysis of this measure.

Value of information

The main drawback of the *competitive analysis* is that it requires to prove the previously stated inequality for all instances of a given problem, which can reveal to be a complex task for real-world applications. The *value of information* proposed by Mitrovic-Minic and Laporte (2004) constitutes a more flexible metric. We will denote by $z_{\mathcal{A}}(I_{of})$ the value of the objective function returned by algorithm $\mathcal{A}$ for the *offline instance* I_{of}. The *value of information* $V_{\mathcal{A}}$ for algorithm $\mathcal{A}$ and instance I is then defined as

$$V_{\mathcal{A}}(I) = \frac{z_{\mathcal{A}}(I_{of}) - z_{\mathcal{A}}(I)}{z_{\mathcal{A}}(I_{of})}$$

Therefore the value of information can be interpreted as the *gap* between the solution returned by an algorithm $\mathcal{A}$ on a instance I and the solution returned by the same algorithm when all information from I is known beforehand.

In contrast with the *competitive ratio*, the *value of information* gives information on the performance of an algorithm based on empirical experience, without requiring optimal solutions for the *offline* instances. It captures the impact of the dynamism on the solution yield by the considered algorithm.

3.4 Benchmarks

To the best of our knowledge, there is presently no *reference benchmark* for dynamic routing problems. It appears that a great proportion of authors create their own instances to test their algorithms. It is worth noting though that various have based their benchmarks on the one designed by Solomon (1987) for static routing (Bent and Van Hentenryck, 2004a,b, Chen et al., 2006, Chen and Xu, 2006, Gendreau et al., 1999). A description of how the original benchmark can be adapted is given in Van Hentenryck and Bent (2006, Chap. 10).

The interested reader is referred to the web page of Dr. G. Pankratz and V. Krypczyk[3] for an updated list of publicly available instances sets for dynamic vehicle routing problems.

[3]url: http://www.fernuni-hagen.de/WINF/inhalte/benchmark_data.htm

Conclusions

Recent technological advances provided the tools for companies to manage their fleet in a real-time fashion. Nonetheless, these new technologies also caused a dramatical increase in the complexity of the fleet management tasks, reinforcing the need for decision support systems adapted to dynamic contexts. Consequently, during the last decade the research community have demonstrated a growing interest for the resulting optimization problem, leading to the creation of a new family of approaches specifically designed to efficiently address dynamism and uncertainty. Analyzing the current state of the art, some directions can be drawn for future research in this relatively new field.

Firstly further work should aim at creating a taxonomy of dynamic vehicle routing problem, possibly by extending existing research on static routing (Eksioglu et al., 2009). This would allow a more precise classification of approaches, quantify similarities between problems, and foster the development of generic frameworks.

Secondly, there is currently no reference benchmark for dynamic vehicle routing problems, and most papers on the subject use custom made instances that do not allow accurate comparison between methods. There is therefore a strong need for the development and publication of benchmarks for the most common dynamic vehicle routing problems, that could be used to make an objective comparative analysis of the solution frameworks available in the literature.

With the advent of multi-core processors on any desktop computer, parallel computing is now made affordable. Specifically tailored algorithms should be developed making use of this technique to reduce the time needed for optimization and provide decision makers with highly reactive tools.

Our review of the existing literature revealed that approximately the two thirds of the work done in the area of dynamic routing ignored the stochastic nature of the problem. As we have seen, it can be due to the unavailability or irrelevance of stochastic data. However, we sense that developing algorithms that make use of this valuable information would improve the fleet performance and reduce operating costs, and should therefore be a priority.

Finally, researchers have mainly focused on the routing aspect of the dynamic fleet management. However, in some applications there is more that can be done to improve performance and user satisfaction. For instance in equipment maintenance services, the call center has a certain degree of freedom in setting an appointment. In other words, it means that the client time windows can be defined, or influenced, by the service company. As a consequence, a system which on top of giving a *yes/no* answer to a client request would be able to suggest convenient times for the company would be of higher interest in such contexts.

Acknowledgements

Financial support for this work was provided by the Region Pays de la Loire (France) and the Centro de Estudios Interdisciplinarios Básicos y Aplicados en Complejidad (CeiBA, Colombia). This support is gratefully acknowledged.

References

Abbott, T., Jones, K., Consiglio, M., Williams, D., and Adams, C. (2004). Small aircraft transportation system, higher volume operations concept: Normal operations. Technical Report TM-2004-213022, NASA, NASA Langley Research Center, NASA LaRC,Hampton VA 23681-2199, USA.

Attanasio, A., Bregman, J., Ghiani, G., and Manni, E. (2007). Real-time fleet management at ecourier ltd. In *Dynamic Fleet Management*, volume 38 of *Operations Research/Computer Science Interfaces*, pages 219–238. Springer US.

Attanasio, A., Cordeau, J., Ghiani, G., and Laporte, G. (2004). Parallel tabu search heuristics for the dynamic multi-vehicle dial-a-ride problem. *Parallel Computing*, 30(3):377–387.

Baldacci, R., Toth, P., and Vigo, D. (2007). Recent advances in vehicle routing exact algorithms. *4OR: A Quarterly Journal of Operations Research*, 5(4):269–298.

Balev, S., Guinand, F., Lesauvage, G., and Olivier, D. (2009). Dynamical handling of straddle carriers activities on a container terminal in uncertain environment - a swarm intelligence approach - . In *Proceedings of the 2009 International Conference on Complex Systems and Applications (ICCSA 2009)*, Le Havre, France. University of Le Havre.

Barcelo, J., Grzybowska, H., and Pardo, S. (2007). Vehicle routing and scheduling models, simulation and city logistics. In *Dynamic Fleet Management*, volume 38 of *Operations Research/Computer Science Interfaces*, pages 163–195. Springer US.

Beaudry, A., Laporte, G., Melo, T., and Nickel, S. (2008). Dynamic transportation of patients in hospitals. *OR Spectrum*, Inpress.

Bent, R. and Hentenryck, P. V. (2005). Online stochastic and robust optimization. In Maher, M., editor, *Advances in Computer Science – ASIAN 2009*, volume 3321 of *Lecture Notes in Computer Science*, pages 286–300. Springer Berlin / Heidelberg.

Bent, R. and Hentenryck, P. V. (2006). A two-stage hybrid algorithm for pickup and delivery vehicle routing problems with time windows. *Computers & Operations Research*, 33(4):875 – 893. Part Special Issue: Optimization Days 2003.

Bent, R. and Van Hentenryck, P. (2004a). Regrets only! online stochastic optimization under time constraints. In *Proceedings of the 19th National Conference on Artificial Intelligence (AAAI-04)*, pages 501–506. AAAI Press.

Bent, R. and Van Hentenryck, P. (2004b). Scenario-based planning for partially dynamic vehicle routing with stochastic customers. *Operations Research*, 52(6):977–987.

Bent, R. and Van Hentenryck, P. (2004c). A two-stage hybrid local search for the vehicle routing problem with time windows. *Transportation Science*, 38(4):515–530.

Bent, R. and Van Hentenryck, P. (2004d). The value of consensus in online stochastic scheduling. In *Proceedings of the 14th International Conference on Automated Planning and Scheduling (ICAPS-04)*. AAAI Press.

Bent, R. and Van Hentenryck, P. (2007). Waiting and relocation strategies in online stochastic vehicle routing. In Veloso, M., editor, *Proceedings of ohe 20th International Joint Conference on Artifical Intelligence (IJCAI-07)*, pages 1816–1821.

Benyahia, I. and Potvin, J. (1998). Decision support for vehicle dispatching using genetic programming. *IEEE Transactions on Systems Man and Cybernetics Part A - Systems and Humans*, 28(3):306–314.

Berbeglia, G., Cordeau, J.-F., and Laporte, G. (2010). Dynamic pickup and delivery problems. *European Journal of Operational Research*, 202(1):8 – 15.

Bertsimas, D. (1988). *Probabilistic combinatorial optimization problems*. PhD thesis, Massachusetts Institute of Technology, Dept. of Mathematics.

Bertsimas, D. and Simchi-Levi, D. (1996). A new generation of vehicle routing research: robust algorithms, addressing uncertainty. *Operations Research*, 44(2):286–304.

Bertsimas, D. and Van Ryzin, G. (1991). A stochastic and dynamic vehicle-routing problem in the euclidean plane. *Operations Research*, 39(4):601–615.

Bieding, T., Goertz, S., and Klose, A. (2009). Online routing per mobile phone: A case on subsequent deliveries of newspapers. In Bertazzi, L., Speranza, M., and VanNunen, J., editors, *Innovations in Distribution Logistics*, volume 619 of *Lecture Notes in Economics and Mathematical Systems*, pages 29–51. International Workshop on Distribution Logistics, Brescia, ITALY, OCT, 2006.

Borodin, A. and El-Yaniv, R. (2005). *Online Computation and Competitive Analysis*. Cambridge University Press, Cambridge.

Branchini, R. M., Armentano, V. A., and Lokketangen, A. (2009). Adaptive granular local search heuristic for a dynamic vehicle routing problem. *Computers & Operations Research*, 36(11):2955–2968.

Branke, J., Middendorf, M., Noeth, G., and Dessouky, M. (2005). Waiting' strategies for dynamic vehicle routing. *Transportation Science*, 39(3):298–312.

Brotcorne, L., Laporte, G., and Semet, F. (2003). Ambulance location and relocation models. *European Journal of Operational Research*, 147(3):451 – 463.

Caramia, M., Italiano, G., Oriolo, G., Pacifici, A., and Perugia, A. (2002). Routing a fleet of vehicles for dynamic combined pick-up and deliveries services. In *Proceedings of the Symposium on Operation Research 2001*, pages 3–5, Duisburg, Germany.

Chang, H., Givan, R., and Chong, E. (2000). On-line scheduling via sampling. In *Proceedings of the Artificial Intelligence Planning and Scheduling (AIPS 2000)*, pages 62–71.

Chang, M., Chen, S., and Hsueh, C. (2003). Real-time vehicle routing problem with time windows and simultaneous delivery/pickup demands. *Journal of the Eastern Asia Society for Transportation Studies*, 5:2273–2286.

Chavan, H. D. (2003). A heuristic approach to solve air taxi scheduling problem. Master's thesis, Virginia Polytechnic Institute and State University.

Chen, H.-K., Hsueh, C.-F., and Chang, M.-S. (2006). The real-time time-dependent vehicle routing problem. *Transportation Research Part E: Logistics and Transportation Review*, 42(5):383 – 408.

Chen, Z. and Xu, H. (2006). Dynamic column generation for dynamic vehicle routing with time windows. *Transportation Science*, 40(1):74–88.

Cheung, B. K. S., Choy, K. L., Li, C.-L., Shi, W., and Tang, J. (2008). Dynamic routing model and solution methods for fleet management with mobile technologies. *International Journal Of Production Economics*, 113(2):694–705.

Christiansen, C. and Lysgaard, J. (2007). A branch-and-price algorithm for the capacitated vehicle routing problem with stochastic demands. *Operations Research Letters*, 35(6):773–781.

Cordeau, J.-F., Laporte, G., Potvin, J.-Y., and Savelsbergh, M. W. (2007a). Transportation on demand. In Barnhart, C. and Laporte, G., editors, *Transportation*, volume 14 of *Handbooks in Operations Research and Management Science*, chapter 7, pages 429 – 466. Elsevier.

Cordeau, J.-F., Laporte, G., Savelsbergh, M. W., and Vigo, D. (2007b). Vehicle routing. In Barnhart, C. and Laporte, G., editors, *Transportation*, volume 14 of *Handbooks in Operations Research and Management Science*, chapter 6, pages 367 – 428. Elsevier.

Crainic, T. G., Gendreau, M., and Potvin, J.-Y. (2009). Intelligent freight-transportation systems: Assessment and the contribution of operations research. *Transportation Research Part C: Emerging Technologies*, 17(6):541–557.

Dantzig, G. and Ramser, J. (1959). The truck dispatching problem. *Management Science*, 6(1):80–91.

Dror, M., Laporte, G., and Trudeau, P. (1989). Vehicle routing with stochastic demands: Properties and solution frameworks. *Transportation Science*, 23(3):166–176.

Eksioglu, B., Vural, A. V., and Reisman, A. (2009). The vehicle routing problem: A taxonomic review. *Computers & Industrial Engineering*, In Press, Corrected Proof:–.

Espinoza, D., Garcia, R., Goycoolea, M., Nemhauser, G. L., and Savelsbergh, M. W. P. (2008a). Per-seat, on-demand air transportation part I: Problem description and an integer multicommodity flow model. *Transportation Science*, 42(3):263–278.

Espinoza, D., Garcia, R., Goycoolea, M., Nemhauser, G. L., and Savelsbergh, M. W. P. (2008b). Per-seat, on-demand air transportation part II: Parallel local search. *Transportation Science*, 42(3):279–291.

Fabri, A. and Recht, P. (2006). On dynamic pickup and delivery vehicle routing with several time windows and waiting times. *Transportation Research Part B: Methodological*, 40(4):335 – 350.

Fagerholt, K., Foss, B. A., and Horgen, O. J. (2009). A decision support model for establishing an air taxi service: a case study. *Journal of The Operational Research Society*, 60(9):1173–1182.

Fiegl, C. and Pontow, C. (2009). Online scheduling of pick-up and delivery tasks in hospitals. *Journal of Biomedical Informatics*, 42(4):624 – 632.

Flatberg, T., Hasle, G., Kloster, O., Nilssen, E. J., and Riise, A. (2007). Dynamic and stochastic vehicle routing in practice. In *Dynamic Fleet Management*, volume 38 of *Operations Research/Computer Science Interfaces*, pages 41–63. Springer US.

Fleischmann, B., Gnutzmann, S., and Sandvoss, E. (2004). Dynamic vehicle routing based on online traffic information. *Transportation Science*, 38(4):420–433.

Flood, M. (1956). The traveling-salesman problem. *Operations Research*, 4(1):61–75.

Gendreau, M., Guertin, F., Potvin, J.-Y., and Taillard, E. (1999). Parallel tabu search for real-time vehicle routing and dispatching. *Transportation Science*, 33(4):381–390.

Gendreau, M., Laporte, G., and Semet, F. (2001). A dynamic model and parallel tabu search heuristic for real-time ambulance relocation. *Parallel Computing*, 27(12):1641 – 1653.

Gendreau, M., Laporte, G., and Séguin, R. (1996). Stochastic vehicle routing. *European Journal of Operational Research*, 88(1):3 – 12.

Ghiani, G., Guerriero, F., Laporte, G., and Musmanno, R. (2003). Real-time vehicle routing: Solution concepts, algorithms and parallel computing strategies. *European Journal of Operational Research*, 151(1):1 – 11.

Ghiani, G., Manni, E., Quaranta, A., and Triki, C. (2009). Anticipatory algorithms for same-day courier dispatching. *Transportation Research Part E: Logistics and Transportation Review*, 45(1):96 – 106.

Godfrey, G. and Powell, W. (2002). An adaptive dynamic programming algorithm for dynamic fleet management, i: Single period travel times. *Transportation Science*, 36(1):21–39.

Goel, A. and Gruhn, V. (2005). Large neighborhood search for rich vrp with multiple pickup and delivery locations. In *Proceedings of the 18th Mini Euro Conference on Variable Neighborhood Search*, Puerto de la Cruz,Tenerife, Spain.

Goel, A. and Gruhn, V. (2008). A general vehicle routing problem. *European Journal of Operational Research*, 191(3):650–660.

Haghani, A. and Jung, S. (2005). A dynamic vehicle routing problem with time-dependent travel times. *Computers & Operations Research*, 32(11):2959 – 2986.

Haghani, A. and Yang, S. (2007). Real-time emergency response fleet deployment: Concepts, systems, simulation & case studies. In *Dynamic Fleet Management*, volume 38 of *Operations Research/Computer Science Interfaces*, pages 133–162. Springer US.

Hansen, P. and Mladenovic, N. (2001). Variable neighborhood search: Principles and applications. *European Journal of Operational Research*, 130(3):449 – 467.

Hentenryck, P. V., Bent, R., and Upfal, E. (2010). Online stochastic optimization under time constraints. *Annals of Operations Research*, 177(1):151–183.

Holmes, B., Durham, M., and Tarry, S. (2004). Small aircraft transportation system concept and technologies. *Journal of Aircraft*, 41(1):26–35.

Hvattum, L. M., Lokketangen, A., and Laporte, G. (2006). Solving a dynamic and stochastic vehicle routing problem with a sample scenario hedging heuristic. *Transportation Science*, 40(4):421–438.

Ichoua, S. (2001). *Problèmes de gestion de flottes de véhicuies en temps réel*. PhD thesis, Universite de Montreal, Montreal, Canada.

Ichoua, S., Gendreau, M., and Potvin, J.-Y. (2000). Diversion issues in real-time vehicle dispatching. *Transportation Science*, 34(4):426–438.

Ichoua, S., Gendreau, M., and Potvin, J.-Y. (2003). Vehicle dispatching with time-dependent travel times. *European Journal of Operational Research*, 144(2):379 – 396.

Ichoua, S., Gendreau, M., and Potvin, J.-Y. (2006). Exploiting knowledge about future demands for real-time vehicle dispatching. *Transportation Science*, 40(2):211–225.

Jaillet, P. and Wagner, M. R. (2006). Online routing problems: Value of advanced information as improved competitive ratios. *Transportation Science*, 40(2):200–210.

Jaillet, P. and Wagner, M. R. (2008). Online vehicle routing problems: A survey. In *The Vehicle Routing Problem: Latest Advances and New Challenges*, volume 43 of *Operations Research/Computer Science Interfaces Series*, pages 221–237. Springer US.

Kenyon, A. and Morton, D. (2003). ,stochastic vehicle routing with random travel times. *Transportation Science*, 37(1):69.

Kilby, P., Prosser, P., and Shaw, P. (1998). Dynamic VRPs: a study of scenarios. Technical Report APES-06-1998, University of Strathclyde, Glasgow, Scotland.

Laporte, G., Louveaux, F., and Mercure, H. (1992). The vehicle-routing problem with stochastic travel-times. *Transportation Science*, 26(3):161–170.

Laporte, G., Louveaux, F., and van Hamme, L. (2002). An integer l-shaped algorithm for the capacitated vehicle routing problem with stochastic demands. *Operations Research*, 50(3):415–423.

Larsen, A. (2001). *The Dynamic Vehicle Routing Problem*. PhD thesis, Technical University of Denmark (DTU).

Larsen, A., Madsen, O., and Solomon, M. (2002). Partially dynamic vehicle routing-models and algorithms. *The Journal of the Operational Research Society*, 53(6):637–646.

Larsen, A., Madsen, O. B., and Solomon, M. M. (2007). Classification of dynamic vehicle routing systems. In *Dynamic Fleet Management*, volume 38 of *Operations Research/Computer Science Interfaces Series*, pages 19–40. Springer US.

Larsen, A., Madsen, O. B., and Solomon, M. M. (2008). Recent developments in dynamic vehicle routing systems. In *The Vehicle Routing Problem: Latest Advances and New Challenges*, volume 43 of *Operations Research/Computer Science Interfaces Series*, pages 199–218. Springer US.

Larsen, A., Madsen, O. B. G., and Solomon, M. M. (2004). The a priori dynamic traveling salesman problem with time windows. *Transportation Science*, 38(4):459–472.

Lehuede, F., Pavageau, C., and Peton, O. (2008). Un systeme d'aide a la decision pour planifier les transports vers les etablissements medico sociaux. In *Handicap*, page 6. IFRATH.

Li, J.-Q., Mirchandani, P. B., and Borenstein, D. (2009a). A lagrangian heuristic for the real-time vehicle rescheduling problem. *Transportation Research Part E: Logistics and Transportation Review*, 45(3):419–433.

Li, J.-Q., Mirchandani, P. B., and Borenstein, D. (2009b). Real-time vehicle rerouting problems with time windows. *European Journal Of Operational Research*, 194(3):711–727.

Lund, K., Madsen, O., and Rygaard, J. (1996). Vehicle routing problems with varying degrees of dynamism. Technical report, IMM Institute of Mathematical Modelling.

Mendoza, J. E., Castanier, B., Guéret, C., Medaglia, A. L., and Velasco, N. (2009). A memetic algorithm for the multi-compartment vehicle routing problem with stochastic demands. *Computers & Operations Research*, In Press.

Mes, M., van der Heijden, M., and van Harten, A. (2007). Comparison of agent-based scheduling to look-ahead heuristics for real-time transportation problems. *European Journal of Operational Research*, 181(1):59–75.

Mitrovic-Minic, S., Krishnamurti, R., and Laporte, G. (2004). Double-horizon based heuristics for the dynamic pickup and delivery problem with time windows. *Transportation Research Part B: Methodological*, 38(8):669 – 685.

Mitrovic-Minic, S. and Laporte, G. (2004). Waiting strategies for the dynamic pickup and delivery problem with time windows. *Transportation Research Part B: Methodological*, 38(7):635–655.

Mladenovic, N. and Hansen, P. (1997). Variable neighborhood search. *Computers & Operations Research*, 24(11):1097 – 1100.

Montemanni, R., Gambardella, L. M., Rizzoli, A. E., and Donati, A. V. (2005). Ant colony system for a dynamic vehicle routing problem. *Journal of Combinatorial Optimization*, 10(4):327–343.

Novoa, C., Berger, R., Linderoth, J., and Storer, R. (2006). A set-partitioning-based model for the stochastic vehicle routing problem. Technical Report 06T-008, Texas State University, 601 University Drive San Marcos, TX 78666.

Novoa, C. and Storer, R. (2009). An approximate dynamic programming approach for the vehicle routing problem with stochastic demands. *European Journal of Operational Research*, 196(2):509–515.

Pisinger, D. and Ropke, S. (2007). A general heuristic for vehicle routing problems. *Computers & Operations Research*, 34(8):2403 – 2435.

Powell, W. (1988). A comparative review of alternative algorithms for the dynamic vehicle allocation problem. In Golden, B. and Assad, A., editors, *Vehicle Routing: Methods and Studies*, pages 249–291. North Holland, Amsterdam, The Netehrlands.

Powell, W. (1996). A stochastic formulation of the dynamic assignment problem, with an application to truckload motor carriers. *Transportation Science*, 30(3):195–219.

Powell, W. (2009). What you should know about approximate dynamic programming. *Naval Research Logistics*, 56(3):239–249.

Powell, W., Bouzaiene-Ayari, B., and Simao, H. (2007). Dynamic models for freight transportation. In Barnhart, C. and Laporte, G., editors, *Transportation*, volume 14 of *Handbooks in Operations Research and Management Science*, chapter 5, pages 285–365. North-Holland.

Powell, W., Sheffi, Y., Nickerson, K., Butterbaugh, K., and Atherton, S. (1988). Maximizing profits for north american van lines' truckload division: A new framework for pricing and operation. *Interfaces*, 18(1):21–41.

Powell, W. and Topaloglu, H. (2003). Stochastic programming in transportation and logistics. *Handbooks in Operations Research and Management Science*, 10:555–636.

Powell, W. and Topaloglu, H. (2005). Fleet management. In Wallace, S. and Ziemba, W., editors, *Applications of Stochastic Programming*, volume 5 of *MPS-SIAM series on Optimization*, chapter 12, pages 185–215. SIAM.

Powell, W. B. (2007). *Approximate dynamic programming: solving the curses of dimensionality*, volume 703 of *Wiley Series in Probability and Statistics*. Wiley-Interscience, Hoboken, New Jersey.

Psaraftis, H. (1980). A dynamic-programming solution to the single vehicle many-to-many immediate request dial-a-ride problem. *Transportation Science*, 14(2):130–154.

Psaraftis, H. (1988). Dynamic vehicle routing problems. In Golden, B. and Assas, A., editors, *Vehicle Routing: Methods and Studies*, pages 223–248. Elsevier Science Publishers B.V.

Psaraftis, H. N. (1995). Dynamic vehicle routing: Status and prospects. *Annals of Operations Research*, 61(1):143–164.

Pureza, V. and Laporte, G. (2008). Waiting and buffering strategies for the dynamic pickup and delivery problem with time windows. *INFOR*, 46(3):165–175.

Regan, A., Mahmassani, H., and Jaillet, P. (1998). Evaluation of dynamic fleet management systems - simulation framework. In *Forecasting, Travel Behavior, And Network Modeling*, number 1645 in Transportation Research Record, pages 176–184. 77th Annual Meeting of the Transportation-Research-Board, WASHINGTON, D.C., JAN, 1998.

Romero, M., Sheremetov, L., and Soriano, A. (2007). A genetic algorithm for the pickup and delivery problem: An application to the helicopter offshore transportation. In *Theoretical Advances and Applications of Fuzzy Logic and Soft Computing*, volume 42 of *Advances in Soft Computing*, pages 435–444. Springer Berlin / Heidelberg.

Ropke, S. and Pisinger, D. (2006). An adaptive large neighborhood search heuristic for the pickup and delivery problem with time windows. *Transportation Science*, 40(4):455–472.

Rousseau, L.-M., Gendreau, M., and Pesant, G. (2002). Using constraint-based operators to solve the vehicle routing problem with time windows. *Journal of Heuristics*, 8(1):43–58.

Secomandi, N. (2000). Comparing neuro-dynamic programming algorithms for the vehicle routing problem with stochastic demands. *Computers & Operations Research*, 27(11-12):1201 – 1225.

Secomandi, N. and Margot, F. (2009). Reoptimization approaches for the vehicle-routing problem with stochastic demands. *Operations Research*, 57(1):214–230.

Shaw, P. (1998). Using constraint programming and local search methods to solve vehicle routing problems. In *Principles and Practice of Constraint Programming – CP98*, volume 1520 of *Lecture Notes in Computer Science*, pages 417–431. Springer Berlin / Heidelberg.

Simao, H., Day, J., George, A., Gifford, T., Nienow, J., and Powell, W. (2009). An approximate dynamic programming algorithm for large-scale fleet management: A case application. *Transportation Science*, 43(2):178–197.

Slater, A. (2002). Specification for a dynamic vehicle routing and scheduling system. *International Journal of Transport Management*, 1(1):29 – 40.

Sleator, D. and Tarjan, R. (1985). Amortized efficiency of list update and paging rules. *Communications of the ACM*, 28(2):202–208.

Smolic-Rocak, N., Bogdan, S., Kovacic, Z., and Petrovic, T. (2010). Time windows based dynamic routing in multi-agv systems. *IEEE Transactions on Automation Science and Engineering*, 7(1):151–155.

Solomon, M. (1987). Algorithms for the vehicle-routing and scheduling problems with time window constraints. *Operations Research*, 35(2):254–265.

Spivey, M. and Powell, W. (2004). The dynamic assignment problem. *Transportation Science*, 38(4):399–419.

Stahlbock, R. and Voss, S. (2008). Operations research at container terminals: a literature update. *Or Spectrum*, 30(1):1–52.

Taillard, E., Badeau, P., Gendreau, M., Guertin, F., and Potvin, J. (1997). A tabu search heuristic for the vehicle routing problem with soft time windows. *Transportation Science*, 31(2):170–186.

Taillard, E. D., Gambardella, L. M., Gendreau, M., and Potvin, J.-Y. (2001). Adaptive memory programming: A unified view of metaheuristics. *European Journal of Operational Research*, 135(1):1 – 16.

Taniguchi, E. and Thompson, R. (2002). Modeling city logistics. *Transportation Research Record: Journal of the Transportation Research Board*, 1790(1):45–51.

Thomas, B. W. (2007). Waiting strategies for anticipating service requests from known customer locations. *Transportation Science*, 41(3):319–331.

Topaloglu, H. and Powell, W. (2006). Dynamic-programming approximations for stochastic time-staged integer multicommodity-flow problems. *INFORMS Journal on Computing*, 18(1):31–42.

Toth, P. and Vigo, D., editors (2002). *The vehicle routing problem*, volume 9 of *SIAM Monographs on Discrete Mathematics*. SIAM Philadelphia.

van Hemert, J. I. and Poutré, J. L. (2004). Dynamic routing problems with fruitful regions: Models and evolutionary computation. In *Parallel Problem Solving from Nature*, volume 3242 of *Lecture Notes in Computer Science*, pages 692–701. Springer Berlin / Heidelberg.

Van Hentenryck, P. and Bent, R. (2006). *Online stochastic combinatorial optimization*. MIT Press.

Verweij, B., Ahmed, S., Kleywegt, A., Nemhauser, G., and Shapiro, A. (2003). The sample average approximation method applied to stochastic routing problems: a computational study. *Computational Optimization and Applications*, 24(2):289–333.

Waters, C. (1989). Vehicle-scheduling problems with uncertainty and omitted customer. *The Journal of the Operational Research Society*, 40(12):1099–1108.

Wilson, N. and Colvin, N. (1977). Computer control of the rochester dial-a-ride system. *Report R77-31, Dept. of Civil Engineering, Massachusetts Institute of Technology, Cambridge, Massachusetts*.

Yang, J., Jaillet, P., and Mahmassani, H. (2004). Real-time multivehicle truckload pickup and delivery problems. *Transportation Science*, 38(2):135–148.

Yang, S., Hamedi, M., and Haghani, A. (2005). Online dispatching and routing model for emergency vehicles with area coverage constraints. In *Network Modeling 2005*, number 1923 in Transportation Research Record, pages 1–8. 84th Annual Meeting of the Transportation-Research-Board, Washington, DC, JAN 09-13, 2005.